

PRESENTATION OF CLAIMS

The pending claims are presented as follows.

1. (Original) A method of compiling a page containing markup text into an application that outputs markup in response to a request from a user, said method comprising:
pre-initializing a static variable of a class to contain the markup text; and
loading the class containing the pre-initialized static variable into a shared, read-only memory.
2. (Original) A method according to claim 1, further comprising:
storing the markup text in a resource file associated with the application.
3. (Original) A method according to claim 1, wherein the step of pre-initializing the static variable includes the step of:
reading the markup text from the resource file; and
initializing the static variable of the class based on the read markup text.
4. (Original) A computer-readable medium bearing instructions that, when executed, cause one or more processors to perform the method according to claim 1.
5. (Original) A method of compiling a page containing markup text into an application that outputs markup in response to a request from a user, said method comprising:
generating instructions for the application, borne on a computer-readable medium, said instructions that, when executed, cause one or more processors to perform the steps of:

in one time the application is executed, loading a class containing a static variable into a shared, read-only memory, said static variable being pre-initialized to contain the markup text; and

in a subsequent time the application is executed, accessing the markup text in the shared, read-only memory.

6. (Original) A method according to claim 5, wherein the class is not loaded into the shared, read-only memory in the subsequent time the application is executed.

7. (Original) A computer-readable medium bearing instructions that, when executed, cause one or more processors to perform the method according to claim 5.

8. (Previously Presented) A computer-readable medium bearing instructions of an application for producing markup based on static markup text, said instruction arranged, when executed, to cause one or more processors to perform the steps of:

in one time the application is executed, loading a class containing a static variable into a shared, read-only memory, said static variable being pre-initialized to contain the markup text; and

in a subsequent time the application is executed, accessing the markup text in the shared, read-only memory.

9. (Previously Presented) A method according to claim 1, wherein:
the markup text includes information to be displayed to a user and an annotation directing a user agent how to render the information to be displayed to the user; and

the markup output by the application includes the annotation.

10. (New) A method according to claim 1, wherein the static variable of a class is an array of characters.

11. (Previously Presented) A method according to claim 5, wherein:

the markup text includes information to be displayed to a user and an annotation directing a user agent how to render the information to be displayed to the user; and
the markup output by the application includes the annotation.

12. (Previously Presented) A method according to claim 5, wherein the static variable of a class is an array of characters.

13. (Previously Presented) A computer-readable medium according to claim 8, wherein:

the markup text includes information to be displayed to a user and an annotation directing a user agent how to render the information to be displayed to the user; and
the markup output by the application includes the annotation.

14. (Previously Presented) A computer-readable medium according to claim 8, wherein the static variable of a class is an array of characters.